

## REMARKS

The Examiner is thanked for granting and participating in the examiner interview of June 6, 2003. It is believed that the interview resulted in a significantly clearer comprehension of both the meaning of the pending claims, and the differences between the invention recited in the claims and the art that is used as the basis for the current rejections. Pursuant to the Examiner's request, the following is an attempt to reduce those clarifications to writing to better assist in the recognition of the patentability of the claimed invention.

## TERM INTERPRETATION

In paragraphs 3-6, the Office Action includes interpretations for various terms used in the application. It is understood that it is often helpful to understand a general term by thinking about a specific example of the general term. For example, it may be helpful to think about a car when a claim refers to a four-wheeled passenger vehicle, even though not all four-wheeled passenger vehicles would qualify as cars. However, it should be noted that, while such examples may be helpful for the purpose of understanding and examining a claim, they do not limit the more general scope of the general terms.

For example, it may be helpful to think of the term "modifier stack" relative to "modifications that are made to objects through the use of a visual rendering application" as stated in the Office Action. However, a modifier stack is not limited to use with "modifications that are made to objects *through the use of a visual rendering application*". A visual rendering system is merely an example of one scenario in which embodiments of the invention may be implemented. For another example, the term "object" is not limited to "being a CAD or visual rendering software representation of a real-world physical object" as stated in the Office Action. As described in the first paragraph of page 4 of the Appeal Brief of November 7, 2002 (Paper

No. 17), an "object" in the context of the claims is, generally, data that is in a format dictated by an application.

## REJECTIONS BASED ON PRIOR ART

### Rejections under 35 U.S.C. §102(e)

In paragraph 10 of the Office Action, Claims 1, 2, 4-10 and 12-17 were rejected under 35 U.S.C. §102(e) as allegedly being anticipated by Lowry et al. ("Lowry"; U.S. Pat. No. 4,864,497). Applicant traverses this rejection.

### IMPORTANT PRELIMINARY POINTS ABOUT CLAIM 1

With respect to Claim 1, before specifically describing distinctions between the claim limitations and the disclosure of Lowry, concepts regarding some elements of the claim warrant general emphasis, as follows.

(1) The source object is generated by the source application, and the target object is generated by translating the source object. Therefore, the source application indirectly dictates the construct or constitution of the target object. Hence, it would be inconsistent with the language of Claim 1 to equate Claim 1's "target object" with something that could not possibly be created by translating a "source object" created by a "source application".

For example, if the source object is data representing a car door in a format dictated by a CAD application, then the target object is data representing a car door in a format dictated by the target application. For another example in the context of a database, such as in Lowry, if the source object is a database record having fields A, B and C, then the target object is a record having fields A, B and C. It is important to note that, if the source object is a database record

having fields A, B and C, it would be improper to equate the target object to a record having fields A, B, C and D, especially where field D is a field that is not supported by the source application.

(2) The limitation recited in Claim 1, "wherein said first modification is not supported by said source application" should not be ignored, for it contributes to the context of the invention embodied in Claim 1. Returning to the CAD-application-to-rendering-application scenario referred to in the specification, an example of this limitation is that the rendering (target) application may be able to represent surfaces and surface textures, whereas the CAD (source) application is unable to represent such surfaces and surface textures. In other words, surface and surface texture rendering functionality is "not supported" by the CAD application. For another example in the context of a database, a "Chinese-enabled" database application (the target application) may be able to interpret and use Chinese characters that a source application (e.g. an inventory control program) simply cannot interpret.

#### LOWRY'S SYSTEM

The portions of Lowry that were relied upon in the Office Action (namely col. 11, lines 40-65; col. 31, lines 29-42; FIG. 5A and FIG. 5B) describe a system in which multiple applications may retrieve data from the same slave database, and store data into the same master database. FIG. 5A is probably the best illustration of this aspect of Lowry's system. Lowry's database appears to have a locking mechanism (including database controller DBC 2920 and a DBC lock 569) to prevent two applications from updating the same field. Finally, Lowry discloses that retrieval programs 580 and 590 for retrieving data from the slave database 520 to

the application programs, and converters 540 and 570 that convert data from the application programs prior to storing the data in the master database 510.

### HYPOTHETICAL

Lowry does not disclose or suggest that one of the applications 530 and 560 are capable of making a modification that is not supported by the other application. However, during the telephone conference, it was explained that even if Lowry did disclose that one of applications 530 and 560 was able to make a modification that is not supported by the other application, Lowry would still fail to satisfy the limitations of Claim 1.

To explain how Lowry does not satisfy Claim 1, repeated reference was made to a hypothetical in which the "source application" was a database application that does not support Chinese characters, and the "target application" was a database application that does support Chinese characters. In this hypothetical, the target application is able to make a modification (i.e. the insertion of Chinese characters into a field of a record) that the source application does not support.

In the following explanation, that same hypothetical shall be used to illustrate that Claim 1 cannot be satisfied by any reasonable reading of Lowry.

### DISCUSSION OF CLAIM 1

As a preliminary matter, it should be noted that the source and target objects recited in Claim 1 are not the same object, because **the target object is translated from the source object**. Thus, while Lowry discloses that the presence of a lock may prohibit one application from writing to a shared resource while another application has a write lock to that resource, such a lock would not prevent one application from modifying the claimed source object while

another application modified the target object. Thus, the source and target objects are, generally, corresponding resources in different applications, rather than a shared resource.

Claim 1 recites, among other things:

**(1) revising said target object in said target application (2) to reflect said second modification to said source object (3) without removing said first modification to said target object.**

Each of the limitations delineated as (1)-(3) must be disclosed in Lowry for Lowry to anticipate Claim 1. In the following discussion, it shall be explained that Lowry does not disclose or suggest the preceding limitations.

The Office Action did not make clear what, in Lowry, was supposed to correspond to the "source object" and "target object" recited in the claims. During the telephone interview, no specific answer was given to what, within Lowry, was considered to be the "source object" and the "target" object of the claims. Therefore, it was explained how Claim 1 was not satisfied regardless of what, within Lowry, was considered to be the "source object" and the "target object". The following is an attempt to reduce in writing that explanation.

**NO POSSIBLE SELECTION OF "SOURCE OBJECT" WILL CAUSE CLAIM 1 TO BE SATISFIED**

Keeping in mind that the original source object dictates the constitution of the original translated target object by way of the translation, three possible "definitions" of the source object are discussed, none of which will cause Claim 1 to be anticipated by Lowry. Specifically, scenarios are discussed in which the source object is equated with:

(1) a field of a record;

(2) an entire record that including a field that is modifiable and supported only by the target application; and

(3) an entire record except for a field that is modifiable and supported only by the target application.

For the purpose of explanation, reference shall be made to the scenario in which the target application supports use of Chinese characters, and the source application does not support the use of Chinese characters.

#### Scenario (1)

In scenario (1), suppose a particular field that is considered the source object and such field is converted from a source application to the Lowry “common data structure” and converted from the common data structure to a corresponding field in a target application (target object). Assume that the target application obtains a lock on that field in the target object, modifies that field by adding Chinese characters, and converts and saves the field back to the Lowry database.

At this point, if the source application modifies that field of the original source record (perhaps from its local store) and converts and saves it back to the Lowry database, (a) the Chinese characters will be overwritten or (b) another separate record, or snapshot, will be created separate from the record or snapshot containing the field with the Chinese characters. Thus, this scenario could not possibly satisfy the limitations:

**(1) revising said target object in said target application (2) to reflect said second modification to said source object (3) without removing said first modification to said target object.**

Lowry does not disclose a target object being revised to contain both modifications to corresponding source and target objects. Even if the source application retrieved the target object/field from the Lowry database to perform its modification on that object/field, the Chinese characters would still be lost upon conversion to the source application because, by definition,

the source application does not support Chinese characters. Thus, the target object will never reflect both the Chinese characters, and the subsequent change made by the source application to the source object.

### Scenario (2)

Scenario 2 has a faulty premise, because it does not make sense to say that the source object is a record with a field that is modifiable and supported only by the target application. This is not possible, because the source object is generated by the source application, and it makes no sense to say that an application generates and modifies an object on the one hand, but does not support the object on the other hand.

In scenario (2), the source object coming from the source application cannot be a record with a field that is modifiable and supported only by the target application (in other words, not supported by the source application), as required by Claim 1. Thus, this scenario is not feasible by virtue of the limitation that **the modification made to the target object is not supported by the source application**. The construct of the original source object that is assumed in this scenario, from which the target object is translated, is just not possible in a database record context such as in Lowry, when constrained to the foregoing limitation.

Furthermore, even if the modification made by the source object were to a version stored in the Lowry database which included the field to which Chinese characters were added (i.e., the target object, contrary to the claim language regarding the “second modification to said source object in said source application”), those characters would be lost when the object is converted from the common data structure of Lowry to the source application prior to making the modification by the source application and converting and storing back into the common data structure. Lowry does not teach or suggest a way to insert the Chinese characters back into the target object, now corresponding to the source-modified version of the object, at the target

application so that the target object contains the modifications to the corresponding objects from both applications.

### Scenario (3)

In scenario (3), the original source object coming from the source application does not include the field that is modifiable and supported only by the target application. However, if the original source object coming from the source application does not include the field that is modifiable and supported only by the target application, then the translated target object that corresponds to such a source object will not have that field either. Thus, the target application is unable to modify a field that doesn't exist, such as a field of a data type that supports Chinese characters. Lowry does not teach or suggest a way to insert a new field into a record, which is supported only by the target application, so that the target object is revised to contain the modifications to the corresponding objects from both applications.

In summary, regardless of what resource is chosen from the database of Lowry to represent the source object in Claim 1, Lowry does not disclose or suggest all of the limitations of the claim. Therefore, Lowry cannot and does not anticipate Claim 1.

Regarding the "snapshot file 525" of Lowry, col. 11, lines 52-55 states that the snapshot file is designed to represent a plurality of versions of master data file 515. Hence, if the Lowry database were used in an attempt to capture both the modification made by the source application and the modification made by the target application, there would be no single version of the master file that includes both of the modifications. Rather, there might be two different versions: one represented in a first snapshot file containing the first modification and one represented in a second snapshot file containing the second modification, with no way of revising the target object to reflect both modifications.



Based on all of the foregoing reasons, Claim 1 is patentable over the Lowry reference. Withdrawal of the rejection of Claim 1 is kindly requested.

Claims 2-8 depend directly or indirectly from Claim 1. Therefore, Claims 2-8 are patentable over the references of record for at least the same reasons as Claim 1. Withdrawal of the rejection of Claims 2-8 is requested.

Independent Claim 9 recites limitations similar to Claim 1, whereby, generally, a sequence of modifications are made to respective corresponding objects in different applications that support different formats and both the modifications are reflected in a single object (the “second object”). Therefore, reasons for patentability based on distinguishing over Lowry as discussed above in reference to Claim 1 are also reasons for the patentability of Claim 9, to the extent applicable. Withdrawal of the rejection of Claim 9 is kindly requested.

Claims 10 and 11 depend from Claim 9. Therefore, Claims 10 and 11 are patentable over the references of record for at least the same reasons as Claim 9. Withdrawal of the rejection of Claims 10 and 11 is requested.

Claims 12 and 13 are computer-readable medium and system claims that correspond to the method of Claim 1. Hence, Claims 12 and 13 are patentable over the cited references of record for at least the same reasons as Claim 1. Withdrawal of the rejection of Claims 12 and 13 is requested.

The citations to Lowry that are relied on in paragraphs 22-25 for the rejection of Claims 14-17 are the same citations relied on for the rejection of Claim 1. As discussed above, the referenced portions of Lowry describe a common database structure, converters, snapshot files, a database controller and locks, none of which are applicable to the inventions recited in each of Claims 14-17. For example, Claim 14 recites the following limitations:

generating a hierarchical structure for organizing one or more properties of a source object being translated to a target object, wherein each level of the hierarchical structure is associated with a property of an object and wherein the source object is associated with a source application and the target object is associated with a target application;

using one or more filter objects to determine a location, within the hierarchical structure, to map the one or more properties of the source object; and

storing the hierarchical structure in a target file, wherein the target file is used by the second application to construct the target object.

Lowry does not disclose or suggest any of these limitations, namely, (1) generation of a **hierarchical structure for organizing object properties**, where each level of the hierarchy is associated with a respective object property; (2) **using filter objects to map source properties into the hierarchical structure**; and (3) **storing the hierarchical structure in a target file for construction of a target object** in a target application. Therefore, Claim 14 is patentable over the references of record and withdrawal of the rejection is requested.

Claims 15-17 depend directly or indirectly from Claim 14 and are patentable at least by virtue of their dependence on Claim 14, based on the foregoing reasons. In addition, each of Claims 15-17 recites additional limitations that further distinguish over the reference of record.

For example, Claim 15 recites use of the filter objects in association with **collection objects to map properties into the hierarchy** based on **comparing a property value with a collection value** of a respective collection object.

For another example, Claim 16 recites use of a **modifier stack for storing modifications of the target object** and **linking the modifier stack with a collection object** and **applying a modification of the modifier stack to the target file via the linked collection object**, thereby constructing the target object.

Based on the foregoing, Claims 14-17 are patentable over the cited reference of record. Withdrawal of the rejection of Claims 14-17 is kindly requested.

Rejections under 35 U.S.C. §103(a)

Paragraph 30 of the Office Action rejects Claims 3 and 11 as allegedly unpatentable under 35 U.S.C. §103(a) over Lowry in view of Baraquet et al. (“Baraquet”; “A Data Front-End for Layered Manufacturing”). This rejection is traversed.

Claim 3 depends from Claim 1 and Claim 11 depends from Claim 9. Lowry is relied on in the rejection for the limitations of the independent Claims 1 and 9, however, it is shown above that Lowry does not disclose or suggest several limitations of these claims. Therefore, Lowry does not support a prima facie case of obviousness with respect to Claims 3 and 11. In addition, Baraquet does not cure the deficiencies of Lowry. Since both of Claims 1 and 9 are patentable over Lowry, it follows that dependent Claims 3 and 11 are patentable over Lowry in view of Baraquet when Lowry is relied on for the same limitations as with the rejection of the parent claims. For these reasons, Claims 3 and 11 are patentable over the cited references of record and withdrawal of the rejection of these claims is requested.

## CONCLUSION

For the reasons set forth above, it is submitted that all of the pending claims (Claims 1-17) are in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

Please charge any shortages or credit any overages to Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

Date: 6/10/03

John D. Henkhaus  
John D. Henkhaus; Reg. No. 42,656

1600 Willow Street  
San Jose, CA 95125  
(408) 414-1080  
Facsimile: (408) 414-1076

### CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Box 1450, Alexandria, VA 22313-1450

on 6/10/03 by Clare C. Lunny

## **CLAIMS APPENDIX**

1     1.     (original) A method for translating objects between applications that use different  
2           formats, the method comprising:  
3           generating a source object in a source application;  
4           translating the source object to a target object in a target application, wherein the  
5                target application has a format that is not supported by the source application;  
6           performing a first modification to the target object, wherein said first modification is  
7                not supported by said source application;  
8           performing a second modification to said source object in said source application; and  
9           revising said target object in said target application to reflect said second modification  
10                to said source object without removing said first modification to said target  
11                object.

1     2.     (original) The method of Claim 1, wherein the step of performing the first  
2           modification to the target object includes the step of performing a type of  
3           modification that cannot be performed using said source application.

1     3.     (original) The method of Claim 1, wherein:  
2           the source application is a Computer Aided Design (CAD) application;  
3           the target application is a rendering application; and wherein  
4           the step of generating the source object in the source application includes the step of  
5                generating a CAD object in said CAD application;  
6           the step of translating the source object to the target object includes the step of  
7                translating the CAD object into a rendering object;  
8           the step of performing the first modification to the target object includes the step of  
9                performing a modification to the rendering object;

10 the step of performing a second modification to said source object includes the step of  
11 performing a modification to the CAD object; and  
12 the step of revising said target object includes the step of revising the rendering object  
13 to reflect the second modification that was made to the CAD object without  
14 undoing the first modification to the rendering object.

1 4. (original) The method of Claim 1, wherein:  
2 the source object is associated with a source geometry and one or more source  
3 properties; and  
4 the step of translating the source object to the target object includes the steps of  
5 translating the source geometry to a target geometry; and  
6 translating the one or more source properties to one or more target properties.

1 5. (original) The method of Claim 1, wherein the step of translating the source object to  
2 the target object includes the step of:  
3 building a mapping based on a translation between the source object and the target  
4 object.

1 6. (original) The method of Claim 5, wherein the step of building the mapping includes  
2 the step of:  
3 constructing a hierarchical tree structure, wherein the hierarchical tree structure is  
4 based on one or more properties associated with the source object.

1 7. (original) The method of Claim 6, wherein  
2 the source object is associated with a source geometry and one or more source  
3 properties; and  
4 the step of constructing the hierarchical tree structure includes the steps of:

5           generating a set of tree objects, wherein the set of tree objects include one or  
6           more filter objects that are based on said source properties;  
7           translating the source geometry to a target geometry; and  
8           inserting said target geometry into said hierarchical tree structure based said  
9           one or more filter objects.

1   8.   (original) The method of Claim 7, wherein the step of generating the set of tree  
2       objects includes the steps of:  
3       translating the one or more source properties to one or more target properties;  
4       generating one or more modifier stacks, wherein the one or more modifier stacks are  
5           based on the one or more target properties; and  
6       inserting the one or more modifier stacks into the hierarchical tree structure.

1   9.   (original) A method for translating objects between applications that use different  
2       formats, the method comprising:  
3       generating a first object in a first application;  
4       translating the first object to a second object in a second application, wherein the  
5           second object has a format that is not supported by the first application;  
6       performing a first modification to the second object in the second application;  
7       performing a second modification to said first object in said first application; and  
8       performing a third modification to the second object based on data generated in  
9           response to said second modification to said first object, wherein said third  
10       modification causes said second object to reflect the second modification that  
11       was made to the first object without undoing the first modification to the  
12       second object.

1 10. (original) The method of Claim 9, wherein the step of performing the first  
2 modification to the second object includes the step of performing a type of  
3 modification that cannot be performed using said first application.

1 11. (original) The method of Claim 9, wherein:  
2 the first application is a Computer Aided Design (CAD) application;  
3 the second application is a rendering application; and wherein  
4 the step of generating the first object in the first application includes the step of  
5 generating a CAD object in said CAD application;  
6 the step of translating the first object to the second object includes the step of  
7 translating the CAD object into a rendering object;  
8 the step of performing the first modification to the second object includes the step of  
9 performing a modification to the rendering object;  
10 the step of performing a second modification to said first object includes the step of  
11 performing a modification to the CAD object; and  
12 the step of performing the third modification to the second object includes the step of  
13 performing a third modification to the rendering object to reflect the second  
14 modification that was made to the CAD object without undoing the first  
15 modification to the rendering object.

1 12. (original) A computer-readable medium carrying one or more sequences of  
2 instructions for translating objects between applications that use different formats,  
3 wherein execution of the one or more sequences of instructions by one or more  
4 processors causes the one or more processors to perform the steps of:  
5 generating a source object in a source application;  
6 translating the source object to a target object in a target application, wherein the  
7 target application has a format that is not supported by the source application;



8 performing a first modification to the target object, wherein said first modification is  
9 not supported by said source application;  
10 performing a second modification to said source object in said source application; and  
11 revising said target object in said target application to reflect said second modification  
12 to said source object without removing said first modification to said target  
13 object.

1 13. (original) A system for translating objects between applications that use different  
2 formats, the system comprising:  
3 a memory;  
4 one or more processors coupled to the memory; and  
5 a set of computer instructions contained in the memory, the set of computer  
6 instruction including computer instructions which when executed by the one  
7 or more processors, cause the one or more processors to perform the steps of:  
8 generating a source object in a source application;  
9 translating the source object to a target object in a target application, wherein  
10 the target application has a format that is not supported by the source  
11 application;  
12 performing a first modification to the target object, wherein said first  
13 modification is not supported by said source application;  
14 performing a second modification to said source object in said source  
15 application; and  
16 revising said target object in said target application to reflect said second  
17 modification to said source object without removing said first  
18 modification to said target object.

1 14. (previously added) A method for translating objects between applications that use  
2 different formats, the method comprising:

generating a hierarchical structure for organizing one or more properties of a source object being translated to a target object, wherein each level of the hierarchical structure is associated with a property of an object and wherein the source object is associated with a source application and the target object is associated with a target application;

using one or more filter objects to determine a location, within the hierarchical structure, to map the one or more properties of the source object; and

storing the hierarchical structure in a target file, wherein the target file is used by the second application to construct the target object.

15. (previously added) The method of claim 14, wherein each of the one or more filter objects is associated with a respective level of the hierarchical structure and associated with one or more collection objects of a set of collection objects, and wherein the step of using one or more filter objects comprises:

determining, for a property of the one or more properties of the source object, a property value from a respective filter object that is associated with the property;

comparing the property value with a respective collection value associated with each of one or more respective collection objects of the set of collection objects that are associated with the respective filter object; and

determining a level within the hierarchical structure to map the one or more properties of the source object, based on the comparing the property value with a respective collection value.

16. (previously added) The method of claim 14, further comprising:

upon a modification of a property of the target object in the target application,  
generating a modifier stack for storing the modification, wherein the property of  
the target object is associated with a respective property of the source object;  
linking the modifier stack with a collection object of a set of collection objects, wherein  
each collection object of the set of collection objects is associated with a  
respective level of the hierarchical structure; and  
applying the modification of the modifier stack to the target file via the linked collection  
object to construct the target object.

17. (previously added) The method of claim 16, wherein each of the one or more filter  
objects is associated with a respective level of the hierarchical structure, comprising:  
upon a modification of a property of the source object in the source application, using a  
filter object of the one or more filter objects to determine a level within the  
hierarchical structure to store the modification;  
applying the modification of the property of the source object to the target file that  
includes the stored hierarchical structure, at the determined level within the  
hierarchical structure; and  
applying the modification of the modifier stack to the target file to construct the target  
object.